Manual de instalación, configuración e integración STORK para Proveedores de Servicio para php

Resumen: Esta es un manual para la instalación, configuración e integración a la plataforma STORK para proveedores de servicio. También integradores de sistemas se pueden beneficiar de este manual. Este manual debe ser leído por administradores de sistemas e integradores de aplicaciones para entorno tecnológíco php

Historial del Documento

Version	*Date*	*Modification reason*	*Modified by*
1.0	1/02/2012	Manual de Instalación	Joaquín Alcalde-Moraño Jensen

Índice

Lista de figuras

Lista de abreviaturas

<Abreviatura>	<Explicación>
STORK	Secure idenTity acrOss boRders linKed
PEPS	Pan European Proxy Server
SP	Service Provider (Proveedor de Servicio)
IDP	Identity Provider (Proveedor de Identidad)

Resumen ejecutivo

Este documento ofrece información detallada sobre como configurar, crear y desplegar en PHP una aplicación para un Proveedor de Servicios (SP) para su uso en la red STORK.

Como es necesaria la existencia de un Apache para desplegar la aplicación SP, el documento comienza dando una información básica sobre el servidor.

Después de oso, se describe qué necesita saber el usuario sobre las posibles configuraciones para su proyecto.

Tras leer este manual, el administrador / integrador debería ser capaz de configurar, crear y desplegar una aplicación que sea capaz de conectarse a la red STORK.

1 Introducción

Este documento está dividido en varios capítulos con el fin de permitir al lector acceder fácilmente a las secciones más relevantes para el escenario específico en el que esté trabajando.

En el próximo capítulo se enseña cómo configurar un servidor Apache para usar el Framework simplesamlphp y donde se desplegará el paquete distribuido.

En el tercer capítulo se da una guía rápida de inicio de la DemoSP-PHP.

En el cuarto capítulo se describe cada configuración necesaria para la DemoSP-PHP.

En el quinto se muestra una típica sesión de ejecución.

En el sexto capítulo se demuestra cómo instalar DemoSP-PHP desde cero.

En el capítulo final se enseña cómo usar la SAML PHP API para generar y validar mensajes SAML.

2 Antes de empezar

Asegúrese de disponer de un servidor Apache.

2.1 Keystore

La aplicación SP usa dos ficheros PEM (clave privada y clave pública) para configurar el certificado para firmar las peticiones SAML y el certificado para incluir en la petición SAML.

2.1.1 Instalando los Certificados

La clave pública y privada del certificado del DemoSP-PHP deben ser colocadas en el directorio cert/ dir (ver sección 4.1.1 sobre configurar el *path*).

2.2 Configuración de Apache

1. Edite el fichero apache que contiene la información referente a virtual hosts
2. Añada el siguiente host virtual a la configuración:

```
Alias /SP /var/simplesamlphp/www/SP/
```

3. Guarde y salga.
4. Reinicie su servidor apache.

Puede cambiar el alias de `/SP` si lo desea. Formará parte de la URL cuando accede al SP PHP. En esta guía consideraremos que el alias es "/`SP`".

3 Inicio rápido

El siguiente procedimiento le ayudará a poner en marcha len unos pocos minutos la aplicación usando una distribución de simplesamlphp. Para más detalles mire en los capítulos 4 y 6.

1. Copiar el contenido de la distribución DemoSP-PHP a `/var/simplesamlphp/`
2. Abrir el fichero `DemoSP-PHP/lib/SAML2/Constants.php`
    a. Editar la propiedad `ASSERTION_URL con el valor` https://**insert.your.ip.here** /SP/return.php
3. Abrir el fichero `DemoSP-PHP/metadata/saml20-idp-remote.php`
    a. Editar la propiedad `SingleSignOnService` con el valor https://**insert.your.country.access url.to.STORK**

Ahora puede abrir su navegador y utilizar la aplicación (vea el capítulo 5).

4 Configurar DemoSP-PHP

4.1 SP

El proyecto DemoSP-PHP ofrece ficheros de configuración que pueden modificarse. En esta sección se explica cada propiedad.

4.1.1 config/authsources.php

El fichero `config/authsources.php` provee las principales configuraciones para el SP.

Creando múltiples entradas ‘identifier’ se hace posible crear más de un Proveedor de Servicios (SP) por máquina.

Key	Description
`‘identifier’`	Identificador utilizado para identificar el SP. Ej: DEMO-SP
`Name`	Nombre del SP
`Certificate`	Nombre del fichero del Certificado del SP
`Privatekey`	Nombre del fichero con la clave privada del SP (en formato PEM)
`privatekey_pass`	Password de la clave privada del certificado del SP
`attributes.NameFormat`	Nombre del formato usado en los atributos
`sign.authnrequest`	Firma del AuthnRequest

Los certificados y claves deben ser situadas en el directorio `cert/`.

Para más información, consulte: http://simplesamlphp.org/docs/1.8/saml:sp

4.1.2 lib/SAML2/Constants.php

Hay dos propiedades localizadas en este fichero que tienen que ver con el SP.

Key	Description
`ASSERTION_URL`	La URL del SP que gestionará las respuestas del PEPS.
`SPID`	El ID del SP en uso, necesario para recibir credenciales alemanas.
`SP_VC_FILE`	El path al fichero de control de versiones generado por la herramienta generadora del control de versiones.

4.2 PEPS (IdP)

Dado que en STORK existe un Pan-European Proxy Service (PEPS) entre el SP y los IdP, desde este punto de vista cada referencia al IdP debe ser entendida como SPEPS.

4.2.1 metadata/saml20-idp-remote.php

Contiene información a cerca del destino del IdP de destino.

Creando múltiples entradas ‘metadata se hace posible crear más de un IdP.

Key	Description
`$metadata['identifier']`	Identificador usaro para identificar al IdP. Ej: LOCAL
`Name`	Nombre del IdP.
`SingleSignOnService`	URL del IdP que gestionará la petición
`certFingerprint`	SHA-1 fingerprint del certificado del IdP
`sign.authnrequest`	Firma del AuthnRequest
`Redirect.validate`	Firma válida en redirecciones

Para más información consulte: http://simplesamlphp.org/docs/1.8/simplesamlphp-reference-idp-remote.

4.3 SAML

El fichero `lib/SAML2/Constants.php tiene varias configuraciones sobre el mensaje SAML`.

4.3.1 Atributos

Los arrays `$ids` y `$attrs` contienen los distintos atributos SAML soportados. Si desea añadir un nuevo atributo, tan solo debe añadir una nueva entrada en `$ids` y añadir la consiguiente entrada en `$attrs`:

Key	Description
`'attr'.name`	Nombre del nuevo atributo
`'attr'.uri`	URI del nuevo atributo
`'attr'.nameFormat`	Formato del nombre del nuevo atributo
`'attr'.value`	Valor por defecto del nuevo atributo

4.3.2 País del SP

Si desea añadir nuevos países para autenticaciones o desea modificar el país actual, la propiedad `$spcountries es la indicada`. Simplemente añada una nueva entrada en esa propiedad. No olvide añadir una entrada metadata en `metadata/saml20-idp-remote.php (ver sección` 4.2.1`)`. El identificador del metadata debe tener <u>exactamente el mismo nombre</u> que el identificador añadido en la propiedad `$spcountries`.

4.3.3 Países de los ciudadanos

Si desea añadir un nuevo país de ciudadanos debe modificar la propiedad `$countries`. Simplemente añada una nueva entrada a esa propiedad.

4.3.4 Espacio de nombres (namespaces)

Para ajustar los namespaces y adaptarlos a sus propios requerimientos modifique las siguientes propiedades:

Key	Description
`STORKP_NS`	namespace del protocolo STORK
`SAMLP_NS`	namespace del protocolo SAML
`STORK_NS`	namespace de la assertion STORK

5 Iniciando DemoSP-PHP

Abra su navegador y entre en la siguiente página: “http(s)://**insert.your.ip.here**/SP/”. Ahora debe de estar navegando en la DemoSP- PHP. Para modificar su propio SP refiérase a la siguiente sección.

Debería ver una página similar a la Figura 1.

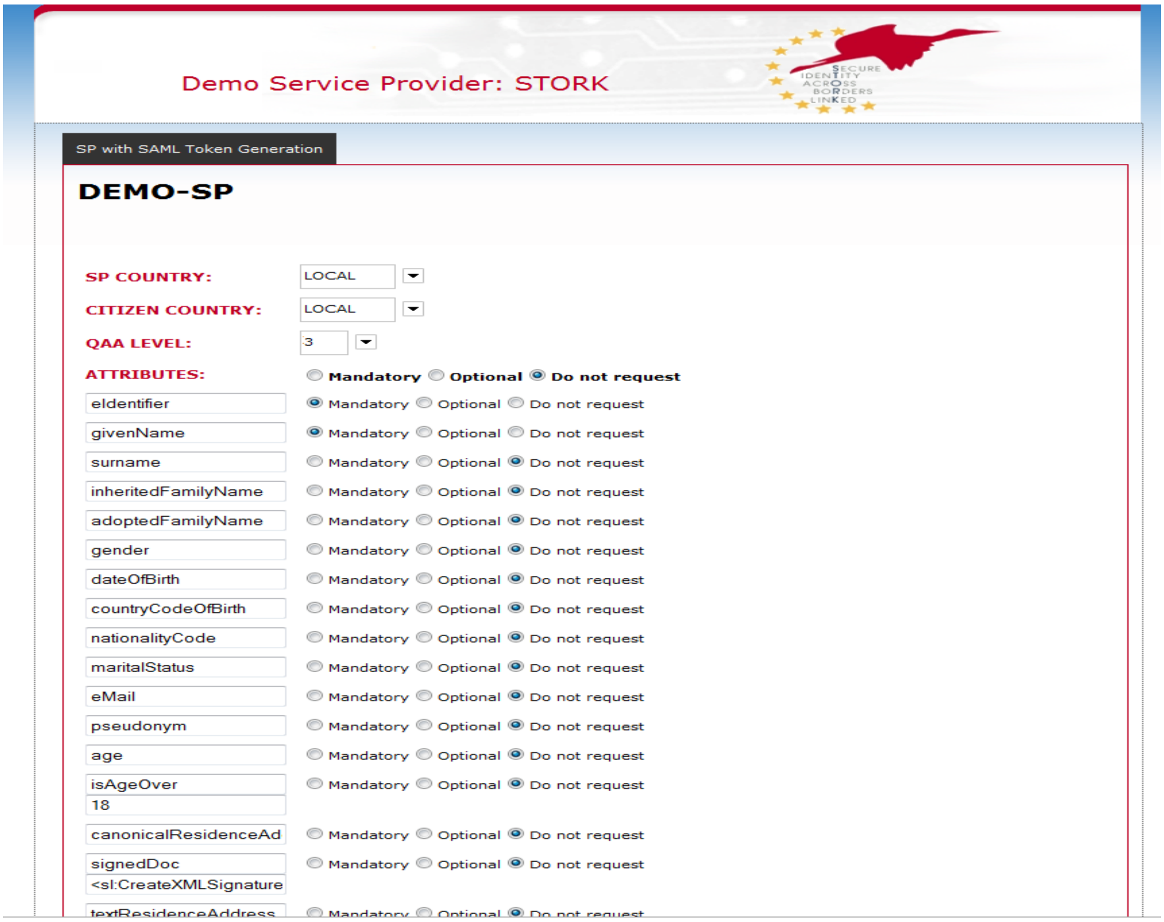


Figura 1 – Página de Inicio - DemoSP:php

Seleccione los atributos que desea solicitar en el proceso de autenticación y pulse “Send”.

Cuando un ciudadano accede al programa STORK, el proveedor de servicio hará una petición de datos del usuario, sobretodo cuando se trata de la primera vez que accede. Estos datos se extraen de sus credenciales o de bases de datos verificadas y mantenidas por las autoridades competentes, de tal manera que el proveedor de servicios pueda fiarse totalmente de los datos recibidos. Además, la calidad de estos datos está ligada al nivel de garantía de calidad de las credenciales requeridas por el proveedor de servicios; algunos de ellos pueden solicitar unos datos de calidad alta, mientras que otros se conformarán con un nivel medio o más bajo.

El proveedor de servicios se basa en los resultados obtenidos de la autenticación online para establecer la identidad de un subscriptor/usuario para realizar la transacción. El proveedor de servicios y el verificador pueden ser la misma entidad o pueden ser entidades diferentes. Si son entidades diferentes, el proveedor de servicios recibe una confirmación por parte del verificador.

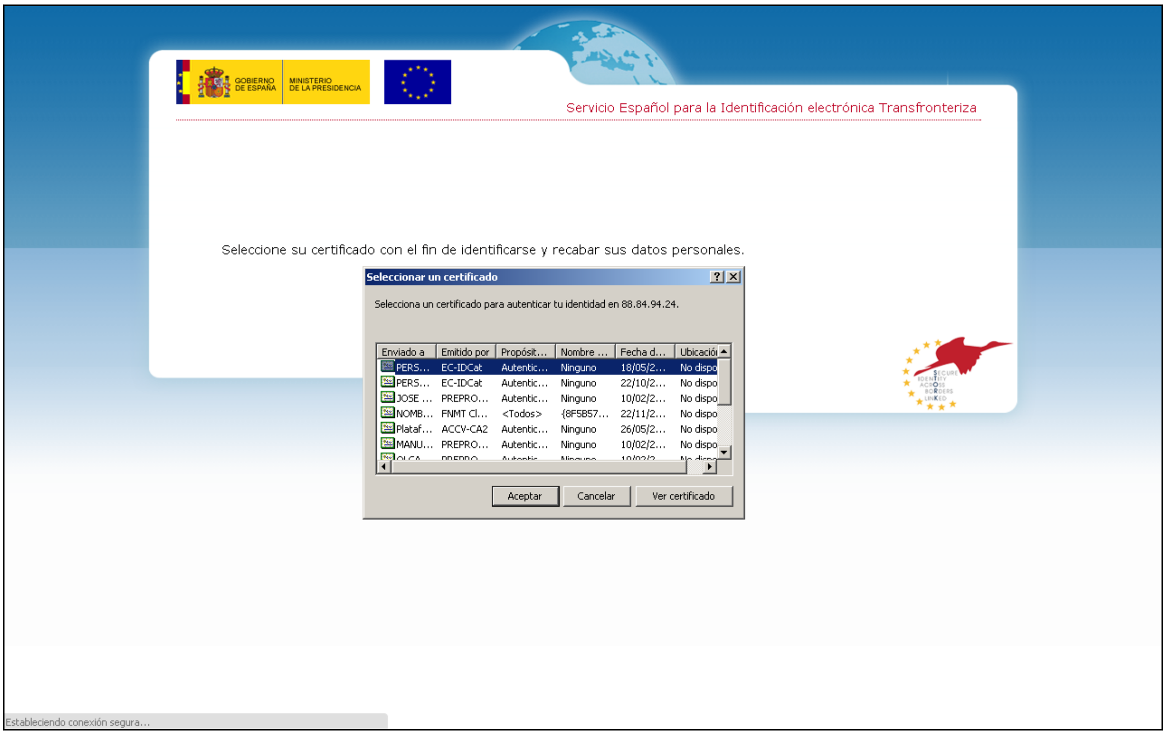
El Proveedor de Servicios es responsable de validar que la confirmación vino de un verificador de confianza. Cuando estas confirmaciones indican la fecha de su creación o atributos asociados al demandante, el Proveedor de Servicios es también responsable de verificar esta información.

El Proveedor de Servicios determina que credenciales son requeridas para proporcionar al demandante o subscriptor acceso. Es por lo tanto el Proveedor de Servicios el que determina el nivel de autenticación para acceder a los datos.

Examinemos el siguiente ejemplo que detalla el funcionamiento del flujo de una petición:

Si un usuario desea acceder a un Proveedor de Servicios español, el usuario comienza el proceso de autenticación seleccionando “autenticarse” con el proveedor de servicios. El proveedor de servicios envía entonces una petición de autenticación y la información relativa al nivel de la QAA (Quality Authentication Assurance) requerida al PEPS español para que proceda con la verificación del demandante. El PEPS español pregunta entonces al usuario qué país le expidió la Identidad electrónica que va a utilizar para autenticarse y le proporciona un listado de Estados Miembros. El usuario selecciona uno de este listado; de acuerdo con esta selección, tanto la demanda de autenticación como el nivel de servicio de QAA requerido son enviados al país seleccionado por el usuario.

El país del ciudadano proporciona al usuario un listado de Identidades electrónicas que cumplen con la autenticación demandada y con los requisitos del nivel de servicio de QAA. El usuario escoge en ese momento la Identidad electrónica con la que quiere autenticarse.



La validación de la Identidad electrónica seleccionada se lleva a cabo basándose en la interacción existente entre el país y su ciudadano. Se pueden dar varios casos:

- si no se puede realizar la validación, se informa al usuario y el proceso termina. El fin del proceso consiste en el envío por parte del usuario de un mensaje SAML informando del tipo de error producido.



- si se consigue validar su Identidad electrónica, el país del ciudadano crea una confirmación que es presentada al usuario para que de su consentimiento, paso necesario ya que los datos son considerados como datos personales.



- si el usuario niega su consentimiento, el proceso termina (con el envío del un mensaje SAML al SP informando de la negación del consentimiento).
- si lo da, se envía al PEPS español que a su vez informa al proveedor de servicios, respondiendo la información obtenida sobre el ciudadano.

Si la autenticación termina en éxito, deberá ver una tabla con los atributos pedidos junto con el valor obtenido, como se ve en la Figura. 2.

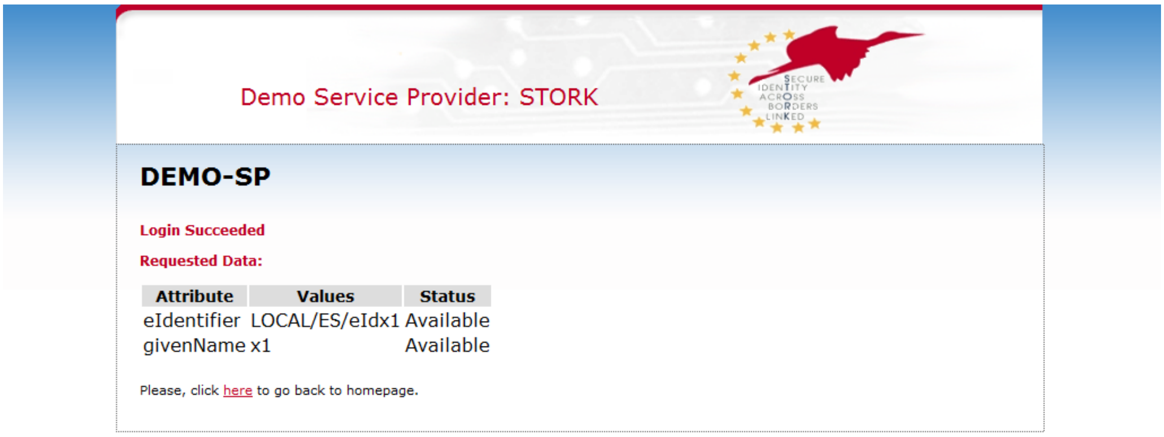


Figura 2 – Página de Retorno - DemoSP php

El fichero de control de versiones para SPs version control file está disponible aquí: "http(s)://your.ip.addres/SP/spInfo.aspx".

sp.local/SP/spInfo.php

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<stork-version-info xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
   <GenerationDate>2011-11-04Z</GenerationDate>
 ▼<countries>
   ▼<country>
      <ID>LOCAL</ID>
      <Name>Local</Name>
    ▼<environments>
      ▼<prod>
          <maxQAA>4</maxQAA>
          <SP-url>http://sp.local:9090/SP/</SP-url>
          <SP-ocsp-url>http://oscp.local:9090/OcspPeps/SPEPS</SP-ocsp-url>
          <Information>Local PEPS</Information>
        ▼<attributes>
            <attribute>eIdentifier</attribute>
            <attribute>givenName</attribute>
            <attribute>surname</attribute>
            <attribute>inheritedFamilyName</attribute>
            <attribute>adoptedFamilyName</attribute>
            <attribute>gender</attribute>
            <attribute>dateOfBirth</attribute>
            <attribute>nationalityCode</attribute>
            <attribute>countryCodeOfBirth</attribute>
            <attribute>maritalStatus</attribute>
            <attribute>canonicalResidenceAddress</attribute>
            <attribute>textResidenceAddress</attribute>
            <attribute>residencePermit</attribute>
            <attribute>eMail</attribute>
            <attribute>age</attribute>
            <attribute>isAgeOver</attribute>
            <attribute>signedDoc</attribute>
            <attribute>citizenQAALevel</attribute>
            <attribute>fiscalNumber</attribute>
            <attribute>title</attribute>
            <attribute>pseudonym</attribute>
          </attributes>
        ▼<certificates>
          ▼<X509Certificate>
              LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tTUlJREp6Q0NBZzhDQkV1b25iSXdEUVlKS29aSWh2Y05BUUVGQlFBd1
            </X509Certificate>
          ▼<UpcomingCertificate>
              <AvailableFrom>2012-05-25+02:00</AvailableFrom>
            ▼<X509Certificate>
                LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tTUlJREp6Q0NBZzhDQkV1b25iSXdEUVlKS29aSWh2Y05BUUVGQlFB
              </X509Certificate>
            </UpcomingCertificate>
          </certificates>
        </prod>
      ▼<test>
          <maxQAA>4</maxQAA>
          <SP-url>http://sp.local:9090/SP/</SP-url>
          <SP-ocsp-url>https://ocsp.local:9090/OcspPeps/SPEPS</SP-ocsp-url>
          <Information>Local PEPS</Information>
        ▼<attributes>
            <attribute>eIdentifier</attribute>
            <attribute>givenName</attribute>
            <attribute>surname</attribute>
            <attribute>inheritedFamilyName</attribute>
            <attribute>adoptedFamilyName</attribute>
```

6 Instando DemoSP-PHP desde cero

En este capítulo se mostrará cómo instalar la DemoSP-PHP desde cero usando el framework simplesamlphp.

1. Descargue la version 1.8.0 de http://code.google.com/p/simplesamlphp/downloads/list
2. Extraiga el paquete descargado a `/var/simplesamlphp/`
3. Abra `/var/simplesamlphp/lib/SAML2/Assertion.php`
    a. En el método `parseAuthnStatement` comente las siguientes líneas:
    ```
    $accr = SAML2_Utils::xpQuery($ac,
    './saml_assertion:AuthnContextClassRef');
        if (empty($accr)) {
    …
    $this->authnContext = trim($accr[0]->textContent);
        }
    ```
    b. En el método `parseAttributes` añada las siguientes líneas:
    ```
    if($attribute->hasAttribute('stork:AttributeStatus')) {
            $this->attributes[$name]['AttributeStatus'] =
    $attribute->getAttribute('stork:AttributeStatus');
          }
    ```
    después:
    ```
      if (!array_key_exists($name, $this->attributes)) {
            $this->attributes[$name] = array();
          }
    ```
    Y modifique:
    ```
    $this->attributes[$name][] = trim($value->textContent);
    ```
    por:
    ```
    $this->attributes[$name]['AttributeValues'] = trim($value->
    textContent);
    ```
4. Abra `/var/simplesamlphp/lib/SAML2/AuthnRequest.php`
    a. En el método `setExtensions` la última línea debería ser:
    ```
    return $root
    ```
5. Copie los fuentes suministrados en DemoSP-PHP-SRC a `/var/simplesamlphp`

7 SAML Engine API

7.1 Generando una petición de autenticación STORK

Existe tres propiedades fundamentales que deben ser creadas antes de la petición SAML: extensions, SP metadata e IdP metadata. Sólo entonces se puede generar el SAML.

```
//el nombre del metadata del SP
$authSource = 'ES-SP';

//cargar el metadata del SP
$as = SimpleSAML_Auth_Source::getById($authSource);
$metadata = $as->getMetadata();

//cargar el metadata del IdP. Para el parámetro 'country' se espera que sea
puesto por el HTML form
$idp = $_POST['country'];
$idpMetadata = $as->getIdPMetadata($idp);

//cargar extensiones
$extensions = StorkConstants::genAttrs($_POST);

//generar una Authentication Request
$ar = stork_saml_Message::buildAuthnRequest($extensions, $metadata,
$idpMetadata);
```

Después de que el SAML Request sea generados se debe realizar un POST hacia el IdP apropiado.

```
$b = new SAML2_StorkHTTPPost($idp);
$b->send($ar);
```

7.2 Validando y leyendo una respuesta de autenticación STORK

Después de recibir un POST se debe validar la firma del SAML.

```
//obtener la response
$b = new SAML2_HTTPPost();
$response = $b->receive();

//obtener el metadata
$authSource = 'PT-SP';
$as = SimpleSAML_Auth_Source::getById($authSource);
$metadata = $as->getMetadata();

//validar la firma
$retVal = stork_saml_Message::checkSign($metadata, $response);
if($retVal) {
    //obtener las assertions
    $assertions = $response->getAssertions();
    //obtener los atributos
    $attributes = $assertions[0]->getAttributes();
    //obtener el status de la saml response
    $status = $response->getStatus();
    if('urn:oasis:names:tc:SAML:2.0:status:Success' !== $status['Code']) {
           // Autenticación correcta!
    } else {
           // Autenticación Fallida!
    }
  } else {
```

```
    // Validación de firma fallida
    echo '<h2>An error occurred';
    echo '<p>The signature validation failed.</p>';
}
```

8 Conjunto de certificados de prueba

Se distribuye junto con la aplicación DemoSP.NET un conjunto de de certificados para facilitar la realización de pruebas transfronterizas.

Nombre del fichero del certificado	País	Contraseña
Ana_Vzorec_02.pfx	Eslovenia	Ana_Vzorec_02
Alice Auth*.p12	Bélgica	BelgaCom.
Stork_Active.p12	Portugal	#ptcert!
sanmiguel.p12	España	1234

9 Preguntas frecuentes

Nota que hay más preguntas frecuentes en el documento “Introducción a STORK para proveedores de Servicios”

¿Dónde puedo acudir a soporte?

mailto:stork@indra.es

10 ANEXO: Petición de Alta como Proveedor de Servicios con acceso a la Plataforma STORK

Fecha_Petición	*Fecha actual*
Nombre	*Nombre del Proveedor de Servicio*
Descripción	
Entorno	*PRE o PRO*
Persona_Contacto	*Persona de Contacto del Proveedor de Servicio (teléfono y email)*
Cerificado_SP	*Parte pública del certificado usado por el Proveedor de Servicio en la firma SAML*
-----BEGIN CERTIFICATE----- -----END CERTIFICATE-----	
Fecha_ prevista	*Fecha de Inicio prevista para pruebas en el entorno indicado*

Nota que nombre del proveedor de servicio y el certificado de firma tienen que coincidir con los valores configurados en 4.1.1.